TMS320C6713搭載
DSPスタータ・キットで学ぶ

# リアルタイムOS「DSP/BIOS」を利用するDSPアプリケーションの開発

三上 直樹

## 第4回 FFTを利用するFIRフィルタ―DSP/BIOSを使わない場合

今回と次回の2回に分けて，前回(2009年5月号，pp.148-159)よりも複雑なマルチスレッド・システムの例として，FFTを利用する2チャネルのFIRフィルタを作成する．このシステムを，今回はDSP/BIOSを使わずに実現し，次回はDSP/BIOSを使って実現する．（筆者）

関連データ

## 1 FFTを利用するFIRフィルタ

FFTを利用するFIRフィルタの処理方法から説明を始めます．説明の都合上，入力信号の長さが確定している場合について最初に説明します．つづいて，今回作るプログラムで必要になる，入力信号の長さが確定していない場合の処理方法について説明します．

### ● 入力信号の長さが確定している場合の処理方法

FFTを使ってFIRフィルタの処理を行う方法は，DFT (Discrete Fourier Transform：離散的フーリエ変換)の循環畳み込み定理に基づいています．

**＜循環畳み込み定理＞**

標本化された信号を$f[n]$，$g[n]$とし，それぞれのDFTを$F[k]$，$G[k]$とします．また，DFTの操作をDFT$\{\cdot\}$で表すものとします．そうすると，次の式が成り立ちます．

$$\mathrm{DFT}\left\{\sum_{m=0}^{N-1} f[m]\cdot g[n-m]_N\right\} = \mathrm{DFT}\left\{\sum_{m=0}^{N-1} f[n-m]\cdot g[m]_N\right\} = F[k]\cdot G[k] \quad \cdots\cdots (1)$$

これが，DFTに関する循環畳み込みの定理です．この式で，$[n-m]_N$は$(n-m)$ modulo $N$[注1]を表すものとします．また，$N$はDFTのデータ数です．

そこで，逆DFTの操作をIDFT$\{\cdot\}$で表すものとすれば，$f[n]$と$g[n]$の循環畳み込みは次の式で計算できます．

$$\sum_{m=0}^{N-1} f[m]\cdot g[n-m]_N = \sum_{m=0}^{N-1} f[n-m]_N\cdot g[m] = \mathrm{IDFT}\{F[k]\cdot G[k]\} \quad \cdots\cdots (2)$$

FIRフィルタの処理は直線畳み込みであって，循環畳み込みそのものではありません．しかし，直線畳み込みになる条件を満足すれば，式(1)と式(2)を使ってFIRフィルタの処理を行えます．

**＜直線畳み込みの計算方法＞**

FIRフィルタに使う入力信号のサンプル数を$L$，係数の個数を$M$，DFTの計算に使うFFTのデータ数を$N$とすると，直線畳み込みになる条件は次のようになります[1]．

$$L+M-1\leq N \quad \cdots\cdots (3)$$

この条件を考慮すると，FFTを使ってFIRフィルタの処理を行う手順は，次のようになります．

①入力信号$x[n]$，$n=0, 1, \cdots, L-1$およびフィルタの係数$h[n]$，$n=0, 1, \cdots, M-1$のそれぞれの後に0を付加して，個数が使用するFFTのデータ数に等しくなったものを$x'[n]$，$h'[n]$とする．ただし，$L$，$M$，$N$は式(3)を満たすように決める．

②$x'[n]$，$h'[n]$のDFTである$X'[k]$，$H'[k]$を，$N$点FFTを使って計算する．

③フィルタの出力信号$y[n]$のDFTである$Y[k]$を，式(4)を使って計算する．

$$Y[k]=H'[k]\cdot X'[k],\ k=0,\ 1,\ \cdots,\ N-1 \quad \cdots\cdots (4)$$

④$N$点逆FFTを使って$Y[k]$の逆DFTである$y[n]$，$n=0, 1, \cdots, N-1$を計算する．この$y[n]$がフィルタの出力信号になる．

ところで，説明したようにDFTおよび逆DFTを使ってFIRフィルタの処理ができますが，直線畳み込みの計算を直接行う方法に比べて，何もメリットがないように見える

注1：整数$n_1$，$n$，$N$に対して，$n_1=n$ modulo $N$とは，$0\leq n_1<N$の範囲の$n_1$に対して，$n=n_1+n_2N$ ($n_2$は整数)という関係が成り立つことである．この説明では直感的にわかりにくいと思う．しかし，$n$が正の整数の場合は簡単で，$n$を$N$で割り算を行ったときの余りが$n_1$であるというのと同じ意味になる．なお，$n$ modulo $N$の代わりに$n$ mod $N$と書く場合もある．